Fig.1

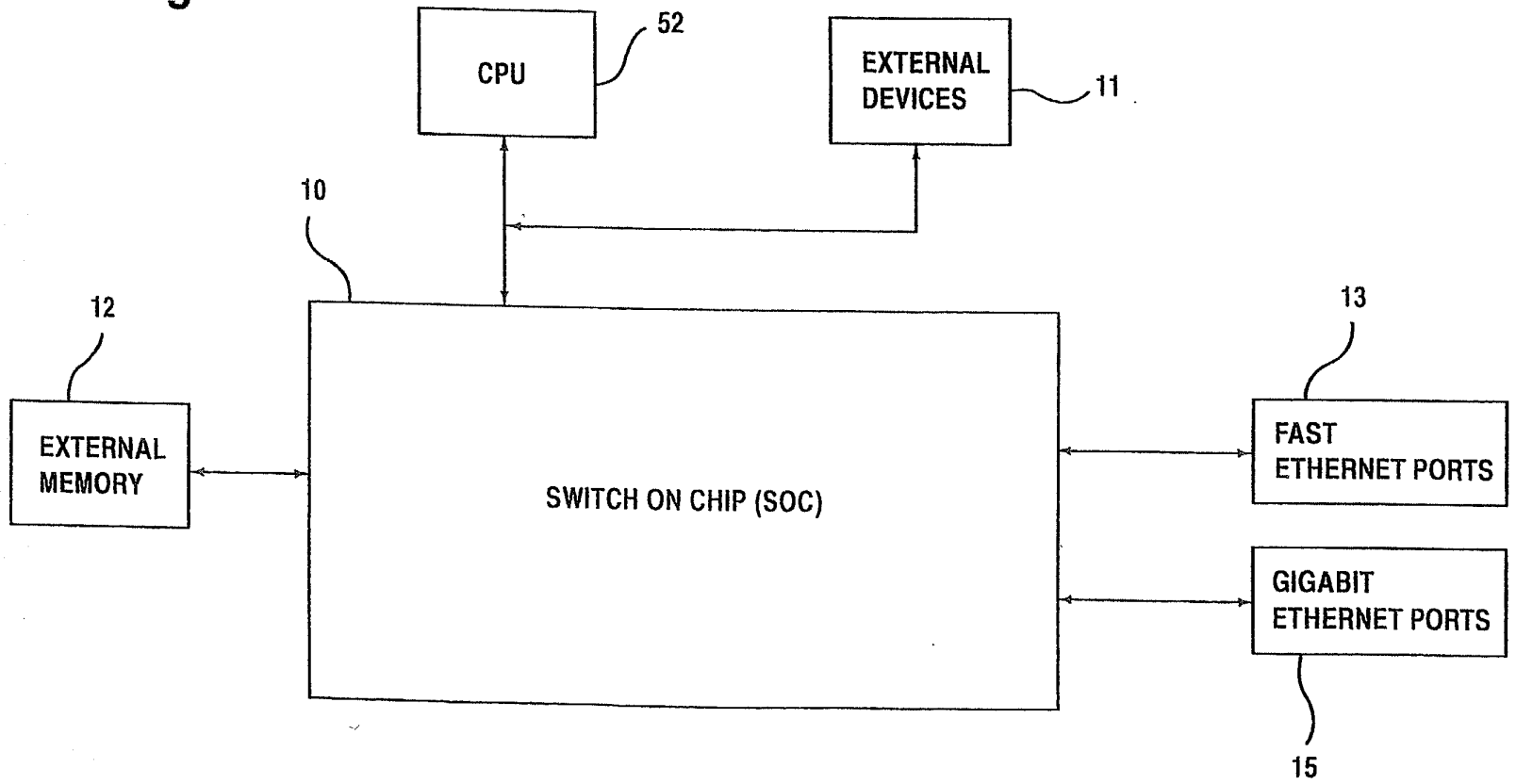

CPU
52
EXTERNAL DEVICES
11
10
12
13
EXTERNAL MEMORY
SWITCH ON CHIP (SOC)
FAST ETHERNET PORTS
GIGABIT ETHERNET PORTS
15

Fig.2

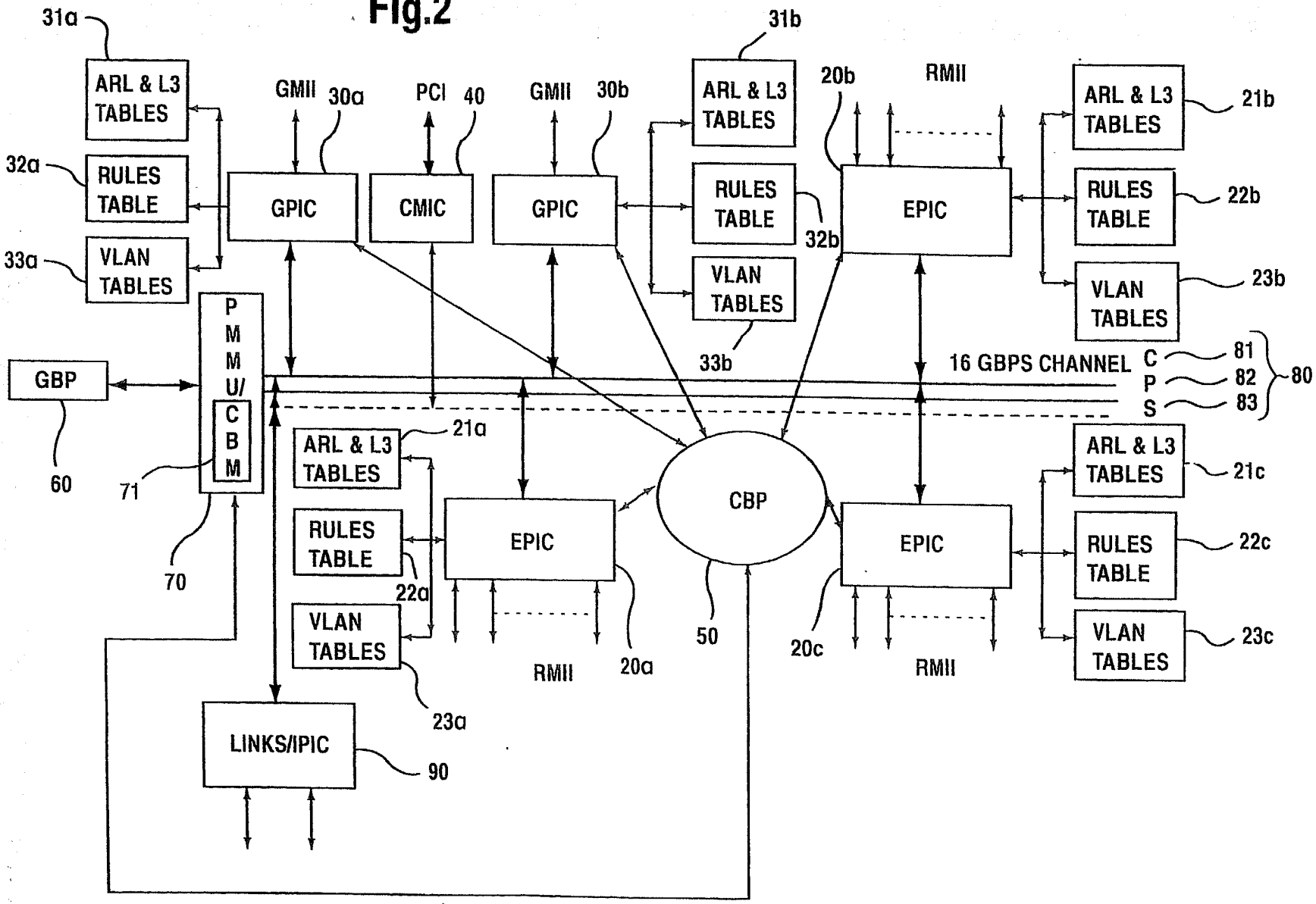

31a
ARL & L3 TABLES
32a
RULES TABLE
33a
VLAN TABLES
GMII
30a
PCI
40
GMII
30b
GPIC
CMIC
GPIC
31b
ARL & L3 TABLES
RULES TABLE
32b
VLAN TABLES
33b
20b
RMII
EPIC
ARL & L3 TABLES
21b
RULES TABLE
22b
VLAN TABLES
23b
16 GBPS CHANNEL
C
P
S
81
82
83
80
GBP
60
PMMU/CBM
71
70
ARL & L3 TABLES
21a
RULES TABLE
22a
VLAN TABLES
23a
EPIC
RMII
20a
CBP
50
EPIC
20c
RMII
ARL & L3 TABLES
21c
RULES TABLE
22c
VLAN TABLES
23c
LINKS/IPIC
90

Fig.3

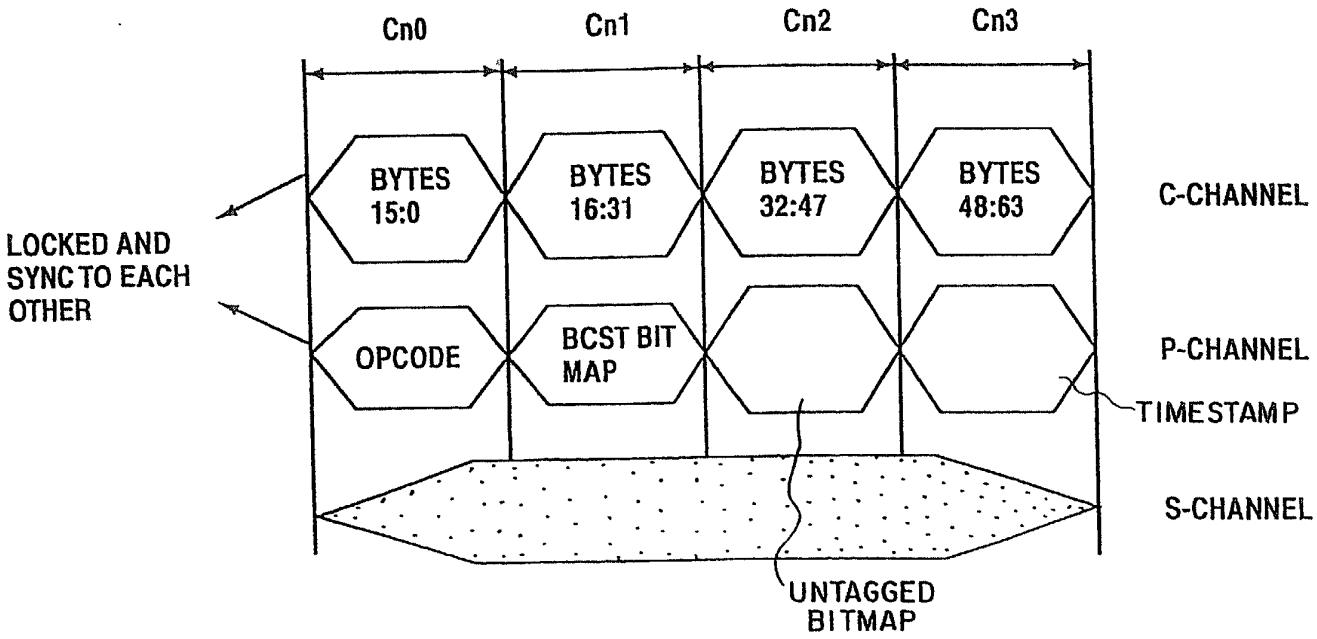

Cn0
Cn1
Cn2
Cn3
BYTES 15:0
BYTES 16:31
BYTES 32:47
BYTES 48:63
C-CHANNEL
LOCKED AND SYNC TO EACH OTHER
OPCODE
BCST BIT MAP
P-CHANNEL
TIMESTAMP
S-CHANNEL
UNTAGGED BITMAP

Fig. 4

PROTOCOL CHANNEL MESSAGES

30	28		26	24	22	20	18	16	14	12		10	8	6	4	2	0
OP CODE	IP	IPX	RESERVED	NXT CELL	SRC DEST PORT			COS	J	S	E	CRC	P	O	LEN		

30	28	26	24	22	20	18	16	14	12	10	8	6	4	2	0
RESERVED		BC/MC PORTBITMAP													

30	28	26	24	22	20	18	16	14	12	10	8	6	4	2	0
U	RES	UNTAGGED PORTBITMAP/SRC PORT NUMBER (BIT0..5)													

30	28	26	24	22	20	18	16	14	12	10	8	6	4	2	0
CPU OPCODES									TIME STAMP						

Fig. 5

SIDE BAND CHANNEL MESSAGES

30	28	26	24	22	20	18	16	14	12	10	8	6	4	2	0
OPCODE			DEST PORT/ DESTINATION DEV ID			SRC PORT			DATA LEN			E	E CODE	COS	O
ADDRESS															
DATA															

Fig. 6

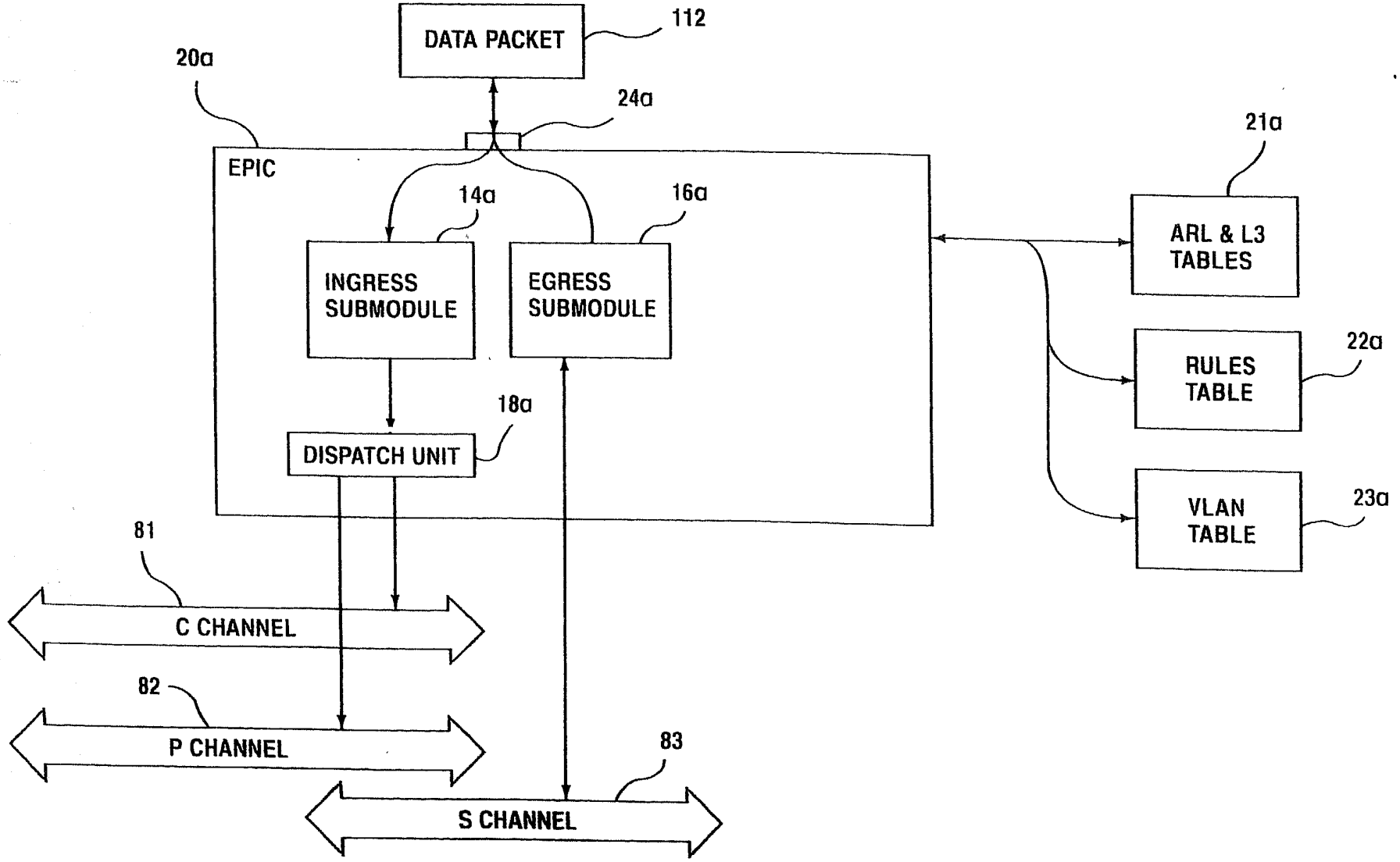


Fig. 7

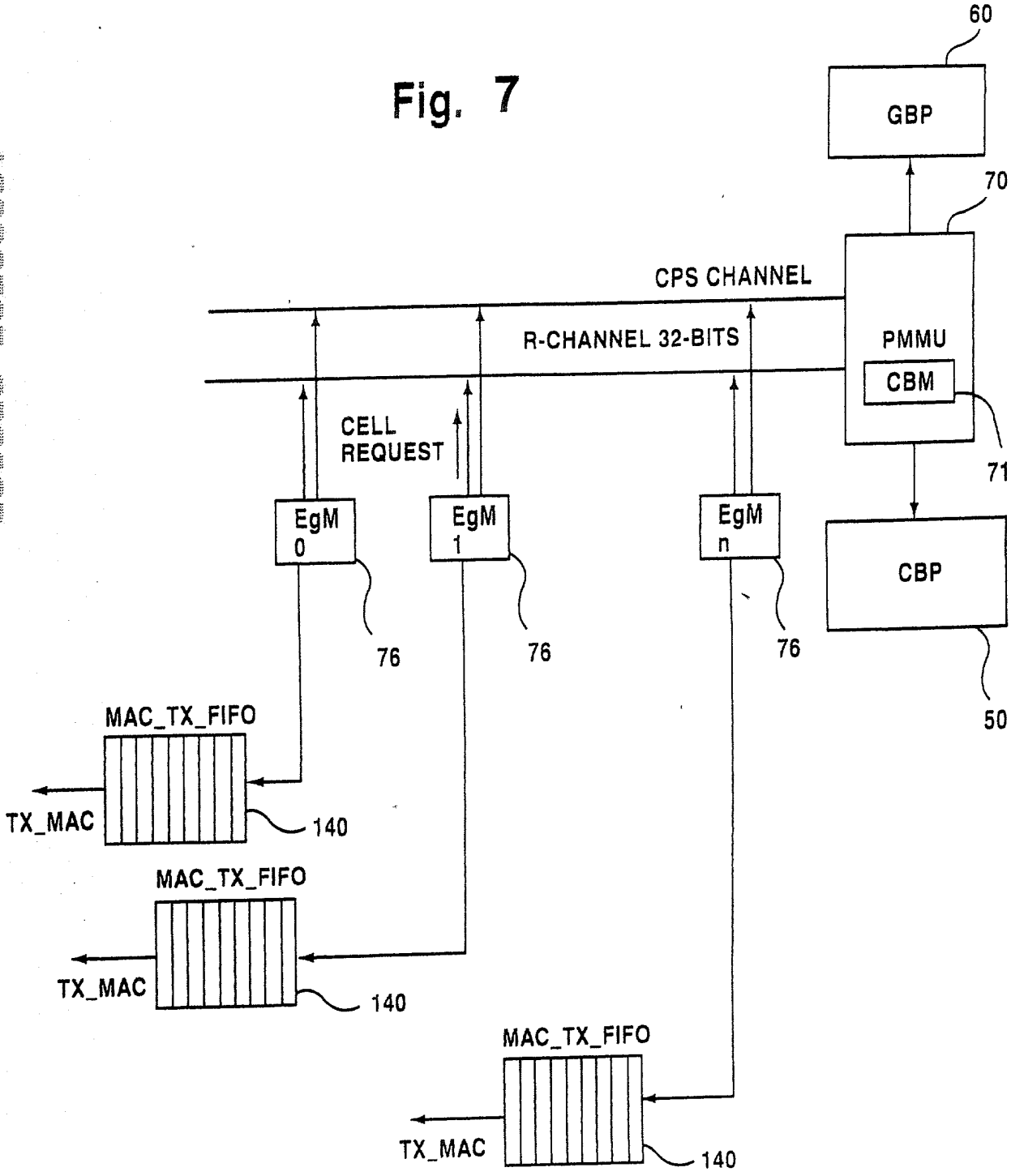

60
GBP
70
CPS CHANNEL
R-CHANNEL 32-BITS
PMMU
CBM
71
CELL
REQUEST
EgM
0
EgM
1
EgM
n
76
76
76
CBP
50
MAC_TX_FIFO
TX_MAC
140
MAC_TX_FIFO
TX_MAC
140
MAC_TX_FIFO
TX_MAC
140

Fig. 8

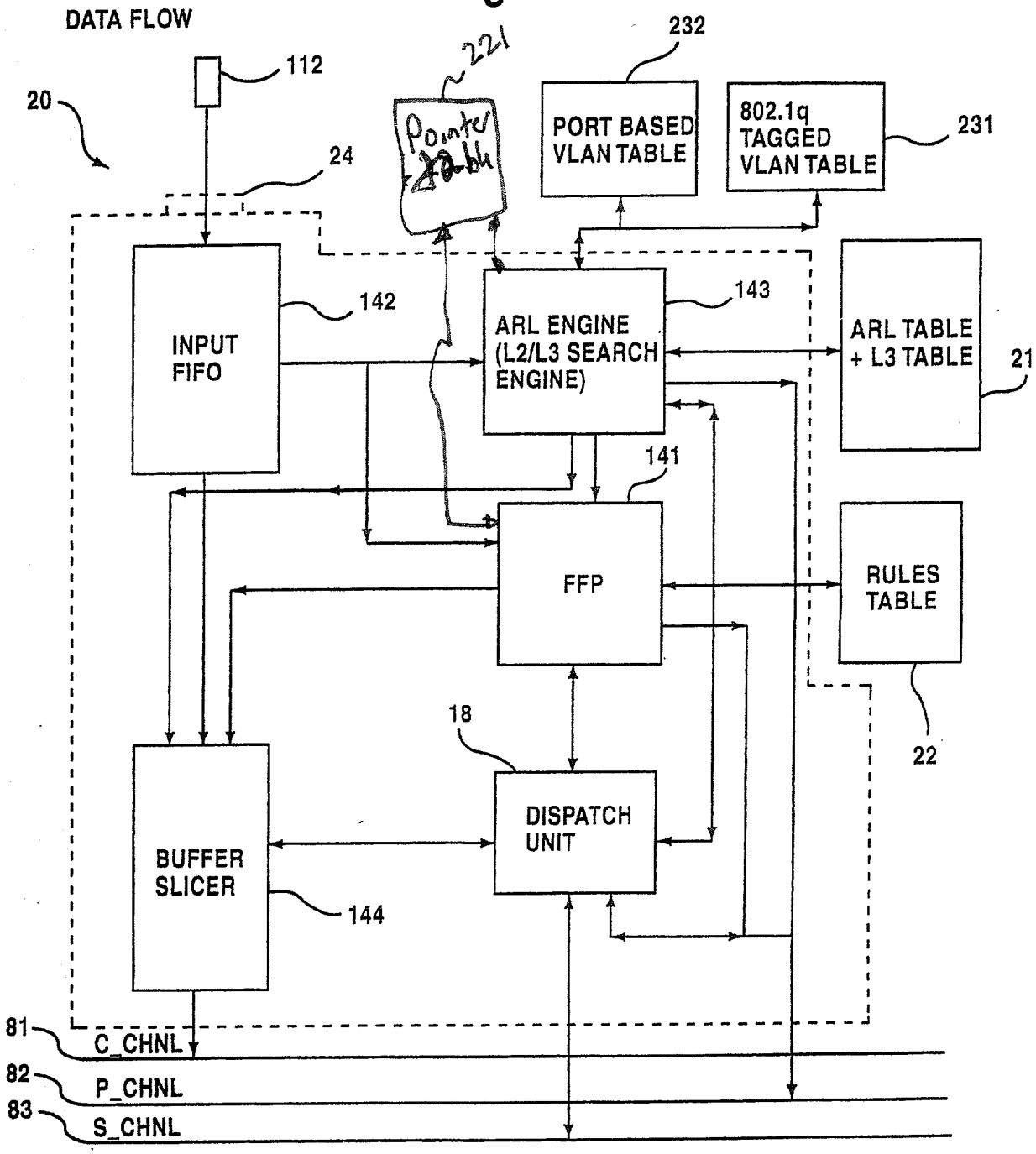

DATA FLOW
20
112
24
221
Pointer
232
PORT BASED VLAN TABLE
802.1q TAGGED VLAN TABLE
231
INPUT FIFO
142
ARL ENGINE (L2/L3 SEARCH ENGINE)
143
ARL TABLE + L3 TABLE
21
141
FFP
RULES TABLE
22
18
DISPATCH UNIT
BUFFER SLICER
144
81
C_CHNL
82
P_CHNL
83
S_CHNL

Fig. 9

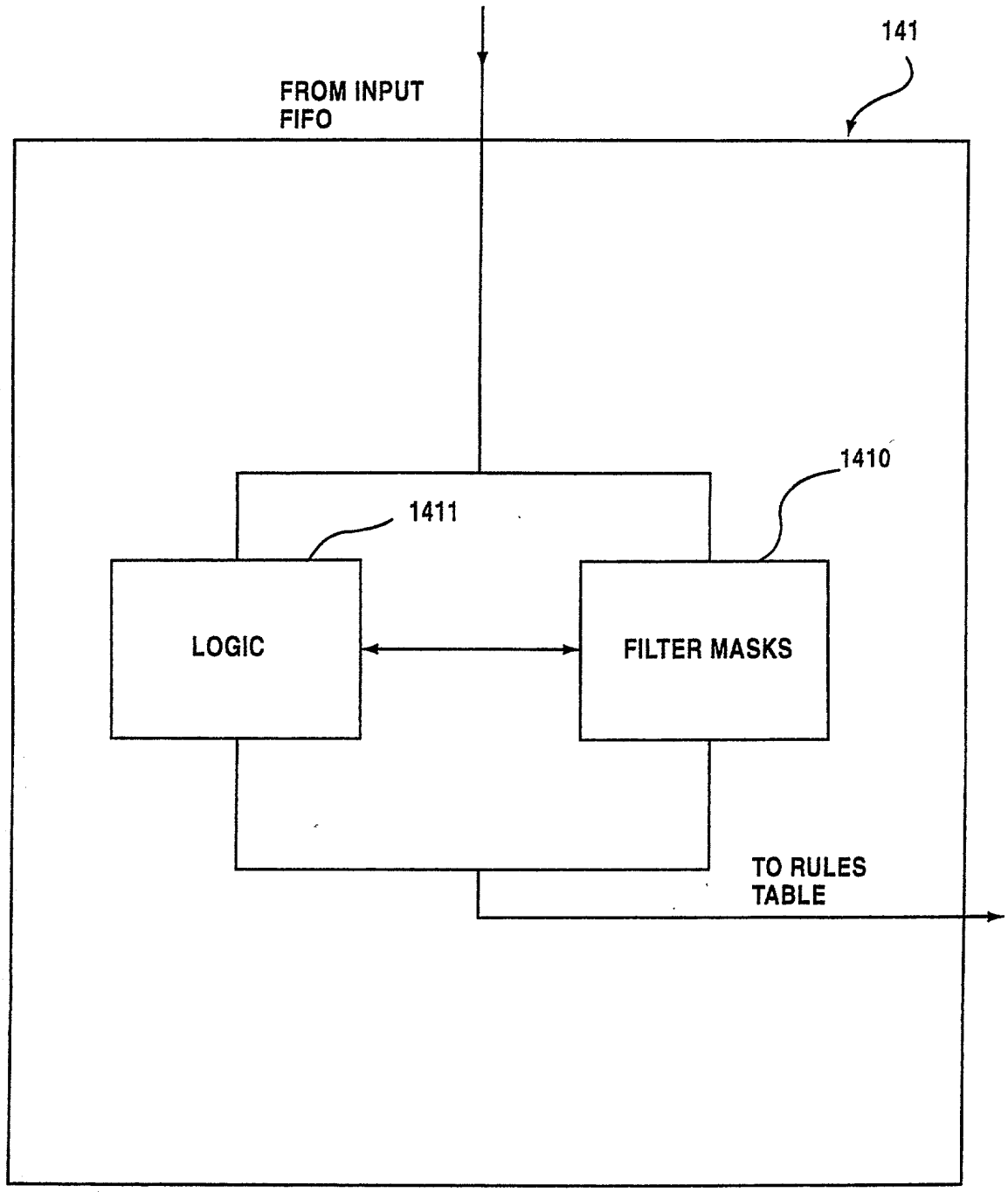
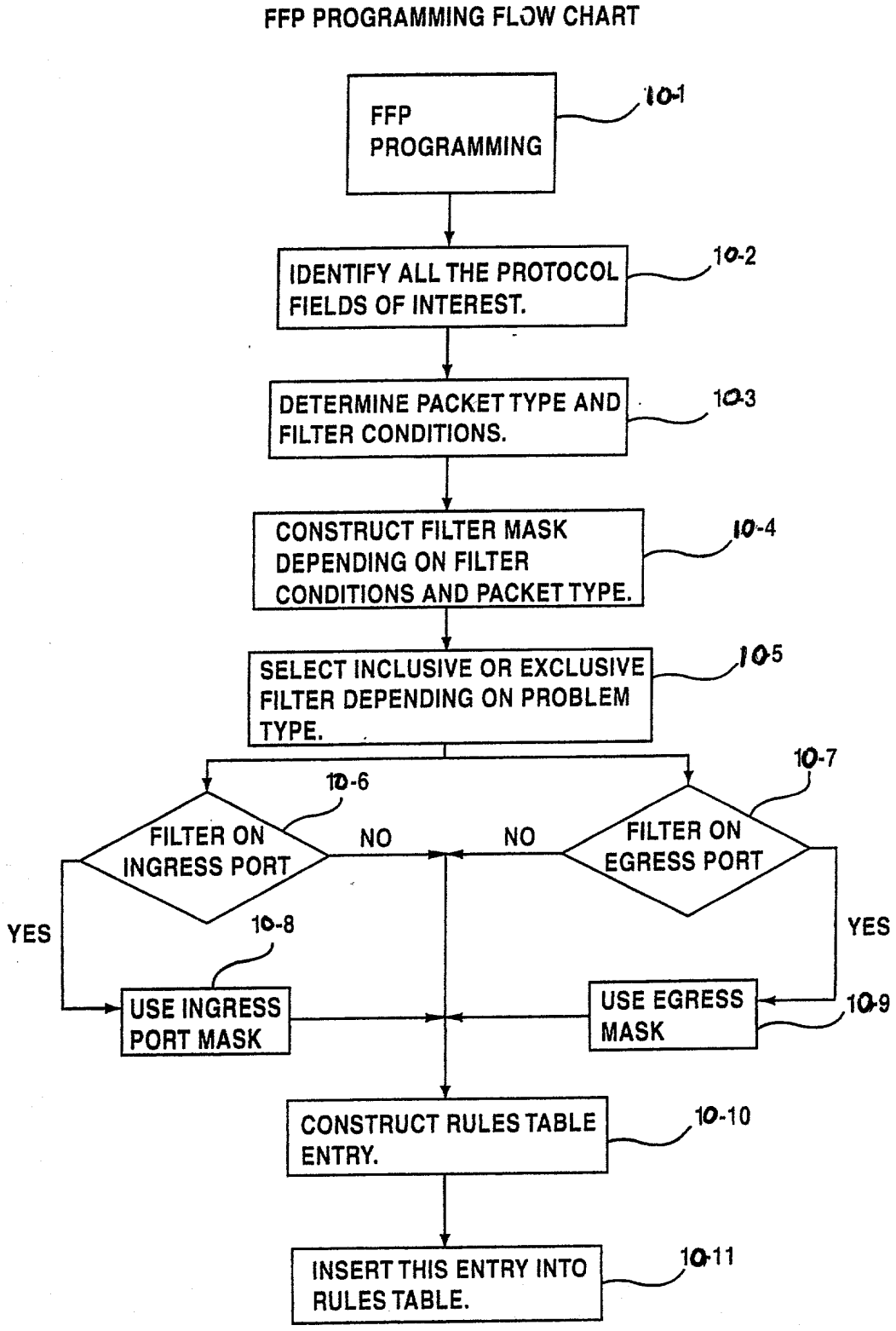

141
FROM INPUT FIFO
1410
1411
LOGIC
FILTER MASKS
TO RULES TABLE

Fig. 10


FFP PROGRAMMING FLOW CHART
FFP PROGRAMMING
10-1
IDENTIFY ALL THE PROTOCOL FIELDS OF INTEREST.
10-2
DETERMINE PACKET TYPE AND FILTER CONDITIONS.
10-3
CONSTRUCT FILTER MASK DEPENDING ON FILTER CONDITIONS AND PACKET TYPE.
10-4
SELECT INCLUSIVE OR EXCLUSIVE FILTER DEPENDING ON PROBLEM TYPE.
10-5
10-6
FILTER ON INGRESS PORT
NO
NO
10-7
FILTER ON EGRESS PORT
YES
10-8
YES
USE INGRESS PORT MASK
USE EGRESS MASK
10-9
CONSTRUCT RULES TABLE ENTRY.
10-10
INSERT THIS ENTRY INTO RULES TABLE.
10-11

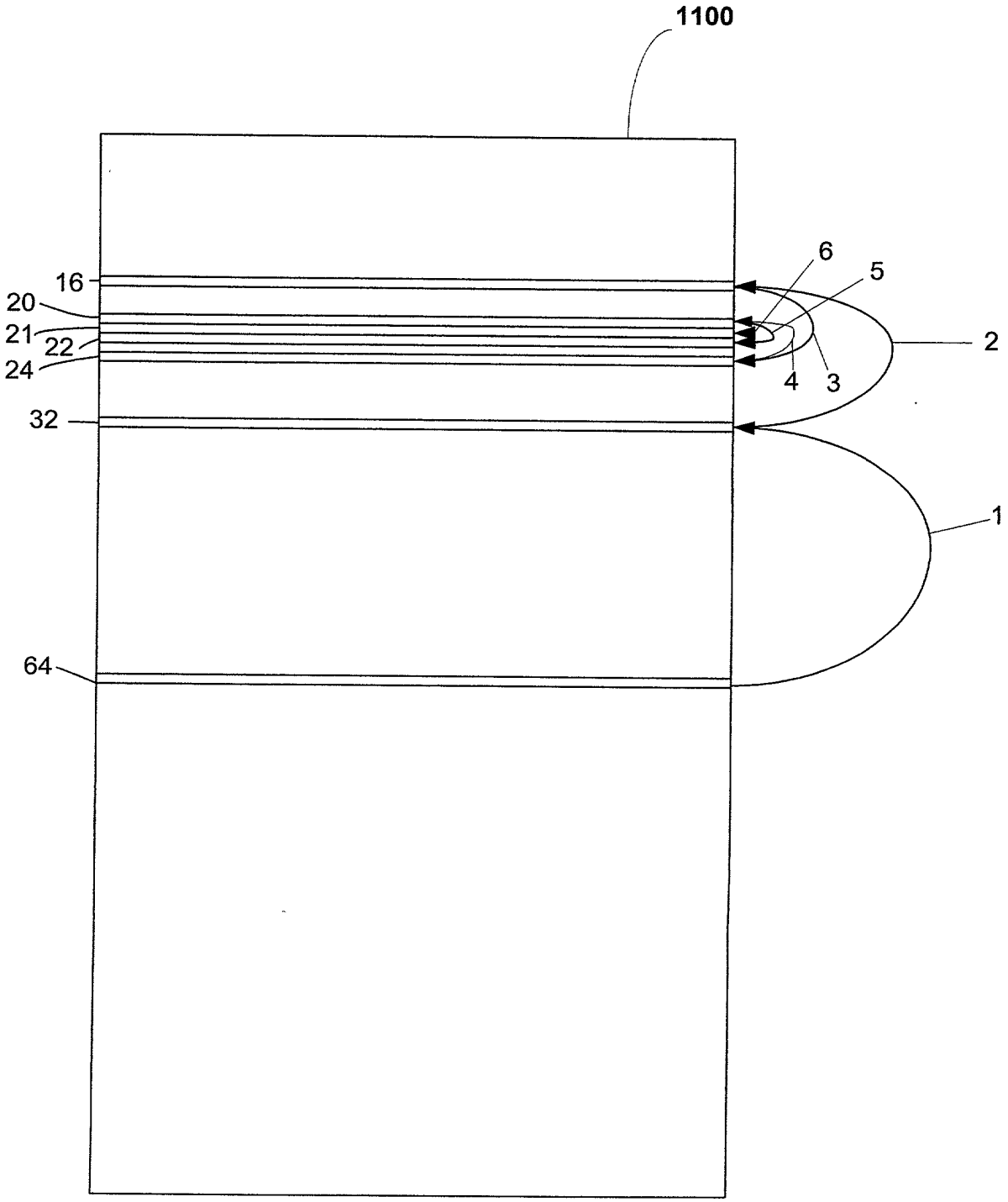


Figure 11

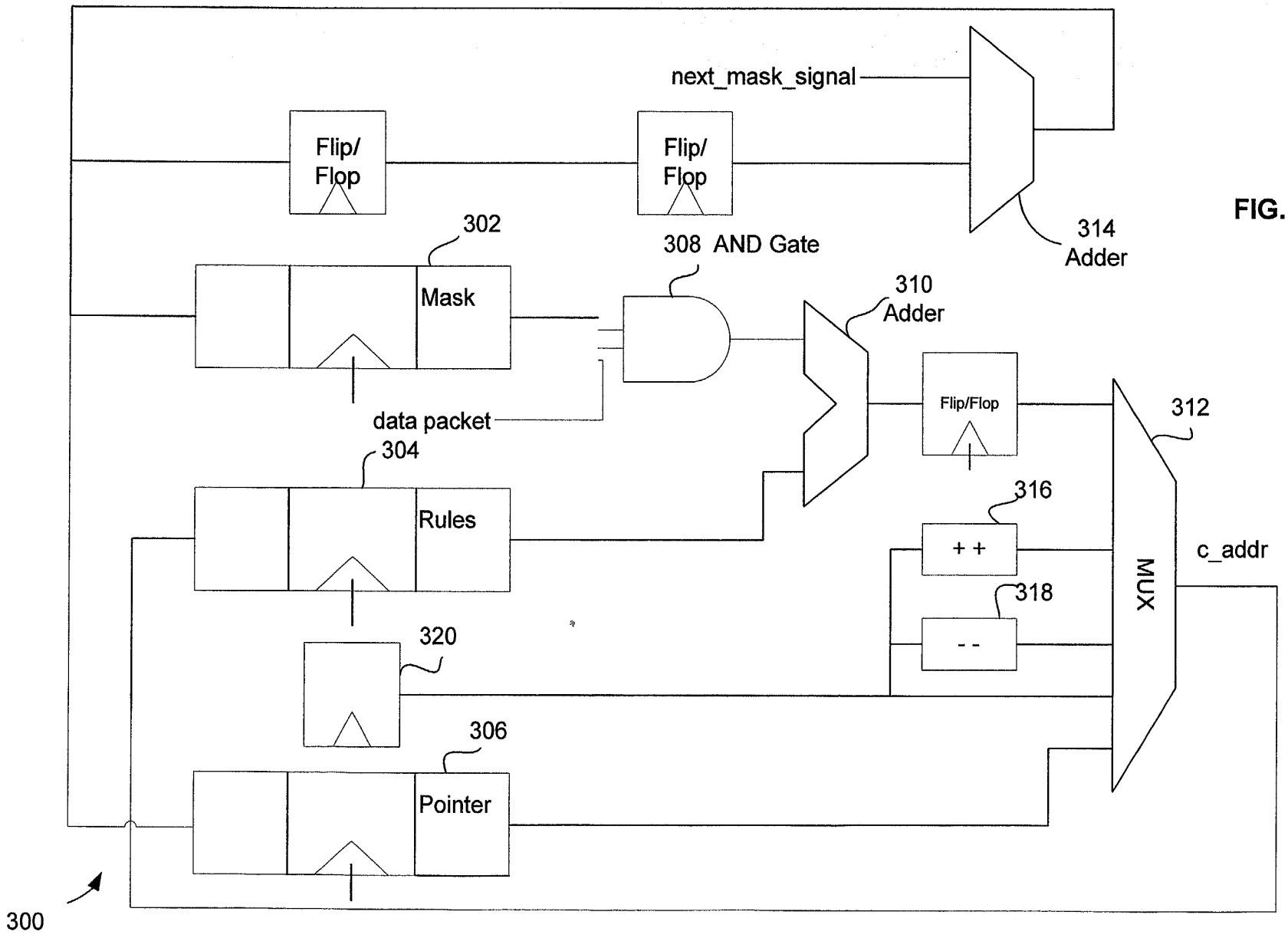

next_mask_signal
Flip/
Flop
Flip/
Flop
314
Adder
302
308 AND Gate
Mask
310
Adder
Flip/Flop
312
data packet
304
316
Rules
+ +
c_addr
318
MUX
- -
320
306
Pointer
300

FIG. 12

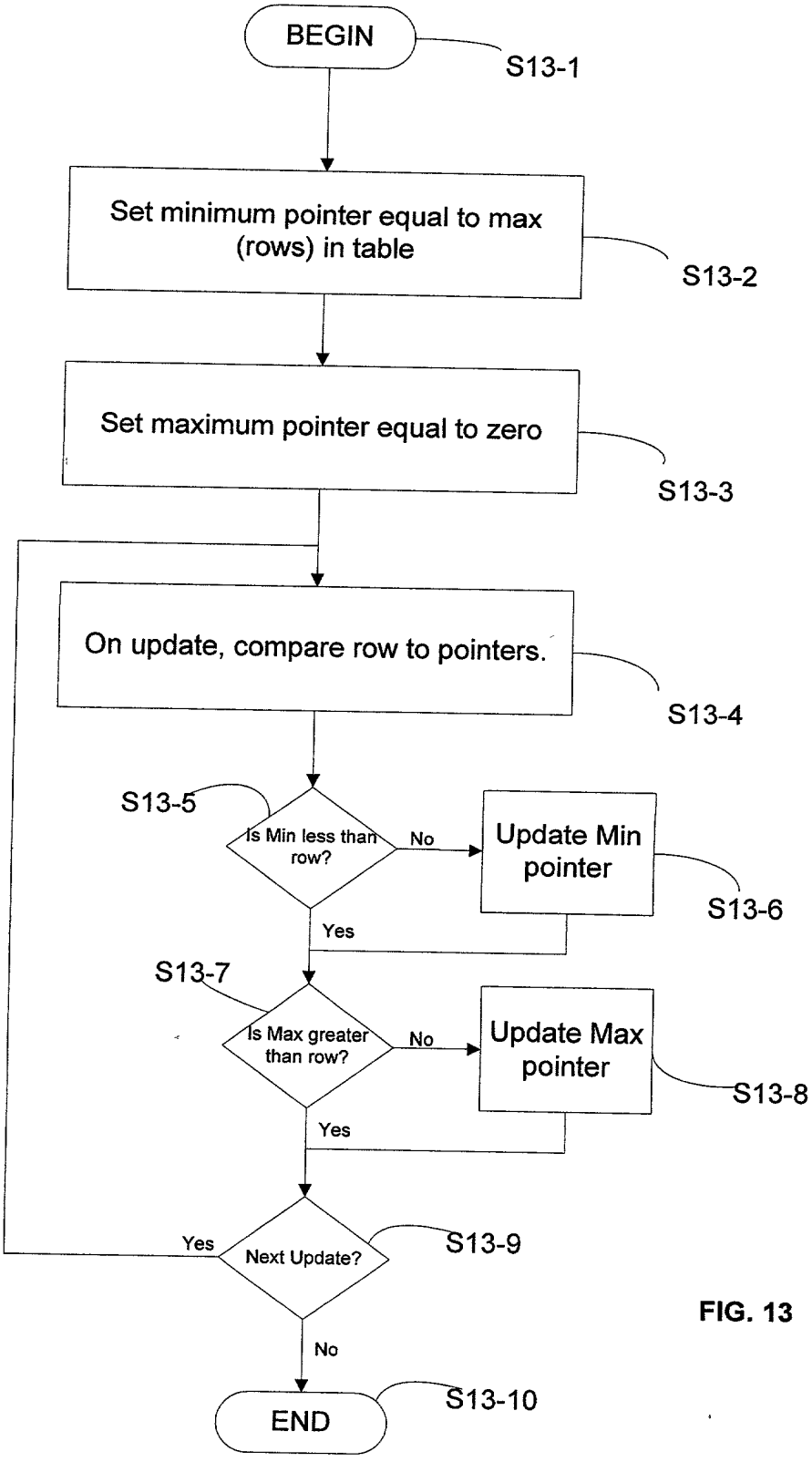

BEGIN
S13-1
Set minimum pointer equal to max (rows) in table
S13-2
Set maximum pointer equal to zero
S13-3
On update, compare row to pointers.
S13-4
S13-5
Is Min less than row?
No
Update Min pointer
S13-6
Yes
S13-7
Is Max greater than row?
No
Update Max pointer
S13-8
Yes
Yes
Next Update?
S13-9
No
END
S13-10

FIG. 13

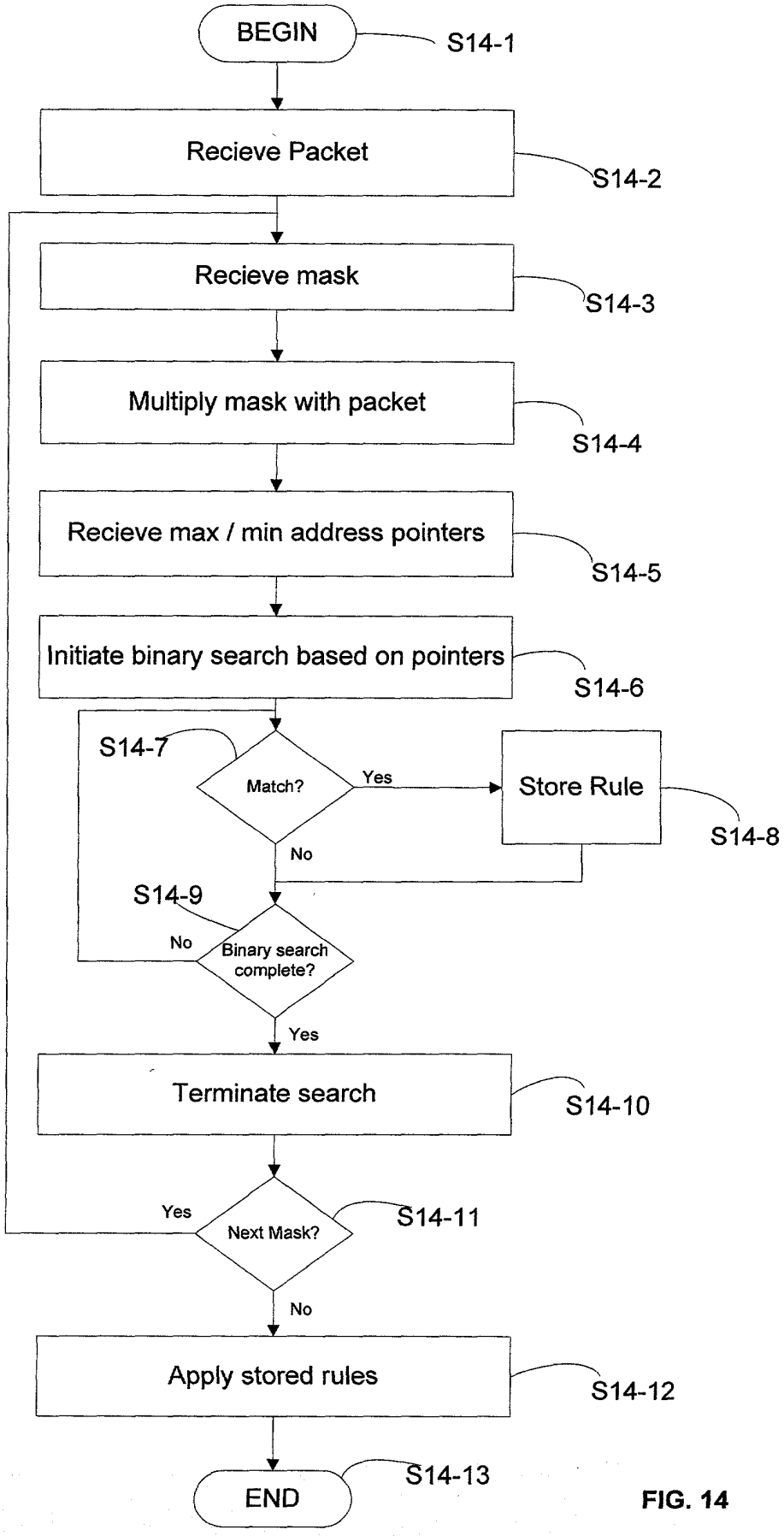

BEGIN
S14-1
Recieve Packet
S14-2
Recieve mask
S14-3
Multiply mask with packet
S14-4
Recieve max / min address pointers
S14-5
Initiate binary search based on pointers
S14-6
S14-7
Match?
Yes
Store Rule
S14-8
No
S14-9
No
Binary search complete?
Yes
Terminate search
S14-10
Yes
Next Mask?
S14-11
No
Apply stored rules
S14-12
END
S14-13

FIG. 14